



Lasse Palokangas

TUOTANNON TESTAUKSEN KEHITTÄMINEN

TUOTANNON TESTAUKSEN KEHITTÄMINEN

Lasse Palokangas
Opinnäytetyö
Kevät 2012
Tietotekniikan koulutusohjelma
Oulun seudun ammattikorkeakoulu

TIIVISTELMÄ

Oulun seudun ammattikorkeakoulu
Tietotekniikka, elektroniikan suunnittelu ja testaus

Tekijä: Lasse Palokangas
Opinnäytetyön nimi: Tuotannon testauksen kehittäminen
Työn ohjaaja: Ensio Sieppi
Työn valmistumislukukausi ja -vuosi: Kevät 2012
Sivumäärä: 45 + 7 liitettä

Tuotannon testaus tarkoittaa tuotannosta tulevan tuotteen tai laitteen testaamista ennen sen siirtymistä asiakkaalle tai seuraavaan tuotantovaiheeseen. Tästä johtuen tuotantotesterin pitää olla nopea ja luotettava, jotta se ei hidastaisi tuotantoprosessia.

Opinnäytetyön aiheena oli tehdä Elektrobit Wireless Communications Oy:lle suunnitelma automaattisen tuotannon testausjärjestelmän testiajan lyhentämiseksi ja testikattavuuden parantamiseksi. Samalla pyrittiin kiinnittämään huomiota laitteen luotettavuuteen ja käytettävyyteen. Työssä keskityttiin oleellisesti testisekvenssin nopeuttamiseen. Myös testitornin luotettavuus ja toimivuus olivat huomion kohteena. Työssä tutustutaan myös valmiin testauslaitteen, kuten Flying Probe -testerin liittämistä osaksi koko testausjärjestelmää.

Työtä tehdessä löydettiin olemassa olevan järjestelmän pahimmat niin sanotut pullonkaulat. Työssä esitetään näihin myös yleistason ratkaisut. Työssä saatiin aikaiseksi suunnitelma testitornista, joka on pienillä muutoksilla toteutettavissa ja selkeästi aikaisempaa kustannustehokkaampi.

Asiasanat:
Tuotannon testaus, testitorni, Flying Probe, piirilevytestaus, neulapeti

ABSTRACT

Oulu University of Applied Sciences
Information Technology, Electronics design and testing

Author: Lasse Palokangas
Title of thesis: The Development of Production Testing
Supervisor: Ensio Sieppi
Term and year when the thesis was submitted: Spring 2012
Pages: 45 + 7 appendices

Production testing means the testing of a manufactured device or product before it is transferred to the client or into the next phase of manufacture. Thus, the production tester must be quick and reliable in order not to slow the production process down.

The idea of my thesis was to conduct a plan for Elektrobit Wireless Communications Oy for shortening the testing time of and improving the test coverage of the automatic production testing system. At the same time, attention was paid on the reliability and usability of the device. The thesis concentrates essentially on accelerating the test sequence. The reliability and functionality of the testing tower were also subjects of attention.

The thesis also introduces incorporating a finished testing device – such as the Flying Probe tester – as a part of the whole testing system.

Furthermore, the worst so-called bottlenecks in the existing system were found in the course of making this thesis. The thesis will also present the generalized solutions for these problems.

Keywords:
Production testing, Flying Probe, PCB-test, bed of nails

ALKULAUSE

Tämän insinööriyön tilaajana toimi Elektrobit Wireless Communications Oy Oulussa. Työn valvojina on Elektrobitin puolesta toiminut Project Managerit Kimmo Väärälä sekä Jarkko Kaippio. Oulun seudun ammattikorkeakoulusta ohjaavana opettajana on toiminut yliopettaja Ensio Sieppi. Tekstin ohjaajana on toiminut lehtori Tuula Hopeavuori.

Haluan kiittää asianosaisia henkilöitä asiantuntevasta työn ohjaamisesta. Kiitokset myös Elektrobitin muulle henkilökunnalle avusta ja vinkeistä työn edistämiseksi.

Erityiskiitos vaimolleni Annikalle tukemisesta insinööriyön tekemisen aikana.

Oulussa 5.4.2012



Lasse Palokangas

SISÄLLYS

TIIVISTELMÄ	3
ABSTRACT	4
ALKULAUSE	5
SISÄLLYS	6
SANASTO	8
1 JOHDANTO	10
2 TUOTANTOTESTAUS	11
2.1 Laadun varmistaminen	11
2.2 Testauksen merkitys	12
2.3 Testauksen luotettavuus	12
2.4 Testauksen kattavuus	12
2.5 Testauskustannukset	13
3 TESTAUSMENETELMÄT	14
3.1 Automaattinen optinen tarkastus	14
3.2 Automaattinen läpivalaisutarkastus	14
3.3 In Circuit Testing ja Flying Probe	14
3.4 Boundary Scan	16
3.5 Toiminnallinen testaus	16
3.6 Built In Self Test	17
4 TESTATTAVIEN TUOTTEIDEN ESITTELY	18
4.1 EB Network Extender	18
4.2 EB Tough VoIP Terminal	19
4.3 EB Tough VoIP Field Phone	19
5 TESTIJÄRJESTELMÄN NYKYTILA	20
5.1 Testitorni	21
5.2 Jigit	22
5.3 Liitännät ja väylät	23
5.4 Ohjelmisto	24
5.5 Testisekvenssin rakenne	24
5.5.1 MSU	24
5.5.2 MCU	27

6 TESTERIN TAVOITETILA	31
6.1 Fyysiset ja mekaaniset muutokset	31
6.2 Ohjelmalliset muutokset	32
6.2.1 Testerin ohjelmointiympäristö	32
6.2.2 Testerin käyttöjärjestelmä	33
6.2.3 MAC-osoite	34
6.2.4 Mittaustulosten käsittely	35
6.3 Testin rakenteen muutokset	35
6.4 Vaihtoehtoiset testausmenetelmät	36
6.5 Testikattavuus	37
6.6 Suunnitelman toteutus	38
7 POHDINTA	40
LÄHTEET	41
LIITTEET	43

SANASTO

BIST	Built-in Self Test, sisäänrakennettu itsetestaus
BS	Boundary Scan
DFT	Design For Testability, tuotteen testattavuuden suunnittelu
DSP	Digital Signal Processor, digitaalisen signaalin käsittely
DUT	Device under test, testattava laite
FP	Flying Probe
GPIO	General Purpose Interface Bus, väylä mittalaitteiden hallintaan
I ² C	Kaksilinjainen sarjaväylä
ICT	In-Circuit Testing, piirin testaus
JTAG	Ohjelmiston ja laitteiston testauksessa käytettävä väylä
Jigi	Tietylle laitteelle suunniteltu testausalusta
NDA	Non-disclosure agreement, salassapitosopimus
MAC	Media Access Control, verkkosovittimen yksilöivä osoite
mBIST	Memory BIST
MCU	EB Tough VoIP Terminalin piirikortti
MFU	EB Tough VoIP Field Phonen piirikortti
MSU	EB Network Extenderin piirikortti
PoE	Power over Ethernet, virta Ethernet-verkosta
PTT	Push-To-Talk
PWB	Printed Wiring Board, piirilevy

SHDSL	Symmetric High-Bitrate Digital Subscriber Loop/Line
UUT	Unit under test, ks. DUT
VoIP	Voice over Internet Protocol

1 JOHDANTO

Työn tarkoituksena oli tuottaa suunnitelma, kuinka tuotannon testausta voidaan kehittää kahden Elektrobit Wireless Communications Oy:n suunnitteleman laitteen piirikortin valmistuksessa. Työssä tutkittiin, miten nykyistä, toiminnassa olevaa testausjärjestelmää voitaisiin nopeuttaa ja sen luotettavuutta parantaa. Työn aloitushetkellä kohteena oleva testerit oli periaatteessa käyttökunnossa, mutta se oli toteutettu nopeasti ja epäsuunnitelmallisesti. Tästä johtuen testerin dokumentoinnit olivat osin puutteelliset, sen ajamat testit olivat hitaita ja niiden luotettavuus oli huono.

Samalla tutkittiin valmiiden järjestelmien, kuten Flying Probe -systeemin, liittämistä osaksi testausjärjestelmää, joko olemassa olevan rinnalle tai sen korvauksiksi järjestelmäksi.

Työhön vaikutti testerin kiireellisyys: testattavien laitteiden tuotantolinja oli tätä testeria vaille valmis. Laitteet oli kehitetty, niiden toiminnallisuus testattu ja kokoonpanolinjakin oli jo kunnossa. Ainoastaan tämä testerit piti saada käytettävyydeltään paremmaksi mahdollisimman pian. Testattavan tuotteen luonteen vuoksi jokainen tuotantolinjalta lähtevä laite pitää testata ennen asiakkaalle toimittamista. Tämä asetti myös insinööriyölle tietyt aikarajat. Oman haasteensa toi myös se, että samalla testerillä pitää pystyä testaamaan vähintään kahta, mahdollisesti jopa kolmea tai neljää eri piirilevyä.

Tässä työssä on käytetty yrityksen sisäisiä dokumentteja, jotka on mainittu liitteinä. Yrityksen NDA:sta johtuen näitä ei kuitenkaan voida liittää varsinaiseen työhön. Näihin viitattaessa pyritään asia silti kertomaan tiedot olennaisilta osin, niin että asia tulee ymmärretyksi.

2 TUOTANTOTESTAUS

Tuotantotestaus on eräs laadunvarmistusprosessi. Laatu kertoo, täyttävätkö luontaiset ominaisuudet vaatimuksia. Laadun varmistamiseksi organisaatiolla kannattaa olla laadunhallintajärjestelmä. Laadunhallinnalla yritys tai organisaatio pyrkii parantamaan omaa toimintaansa. Laadunhallintajärjestelmä kannustaa yritystä tekemään prosesseja, joilla saadaan aikaiseksi tuote, joka on riittävän laadukas asiakkaalle. (1.)

2.1 Laadun varmistaminen

Laadusta on neljä eri näkemystä: tuotantokeskeinen, suunnittelukeskeinen, asiakaskeinen sekä systeemikeskeinen laatu. Tuotantokeskeisessä laadussa tarkastellaan tuotteen virheettömyyttä. Erilaiset kustannukset, kuten tekniset tai taloudelliset, ovat ongelmia, joihin pitää löytää ratkaisut. Näiden kustannusten mittarina toimii siis virheiden määrä. Virheiden lukumäärä voidaan laskea vaikka rikkinäisinä ja korvattuina tuotteina tai tyytymättöminä asiakkaina. Tavoitteena on tietenkin virheettömyys. Tämä on kuitenkin useimmiten saavuttamattomissa ainakin järkevissä kustannusrajoissa. Silloin on tyytyminen siihen, että tuotantolinjalta tulee jonkin verran virheellisiä tuotteita. On kustannustehokkaampaa tuottaa vähän viallisia tuotteita ja korvata ne asiakkaalle, kuin pyrkiä nollatoleranssiin. (2, s. 28–31.)

Tuotantotestaus on merkittävässä roolissa, kun puhutaan tuotteen laadusta suhteessa sen tuotantokustannuksiin. Tuotantotesterillä pyritään vahvistamaan mahdollisimman kattavasti tuotteen toimivuus. Toisinaan on kuitenkin kannattavampaa jättää jokin osa tuotteesta testaamatta, jos sen testaamiseen vaaditaan isoja resursseja. Tuotantotestaus on eräs tällainen laadunvarmistusprosessi, jossa tuotteen laatu pyritään varmistamaan ja mahdolliset vikakohteet löytämään ja korjaamaan jo tuotannon aikana, ennen kuin tuote siirtyy asiakkaalle. (3.)

Elektrobit Oyj:n tuotteet noudattavat ISO 9000 -laatustandardia (ISO 9001:2008 versio). Tämän pohjalta tuotetta on testattava riittävästi riittävällä tarkkuudella, jotta sen luotettavuus ja laadukkuus voidaan verifioida.

2.2 Testauksen merkitys

Testaus on prosessi, jossa arvioidaan tuotetta tai järjestelmää, jotta löydettäisiin mahdolliset virheet ja viat testattavasta tuotteesta. Piireihin lisätään usein testi-logiikkaa, jotta testin kattavuutta saadaan lisättyä tai jotta vähennettäisiin aikaa, joka kuluu tuotteen kelpuuttamiseen. (4, s. 7.)

Tuotantotestauksessa pyritään löytämään viat, jotka ovat syntyneet tuotannossa tai tuotantolinjalla, esimerkiksi virheelliset juotokset tai komponenttiladonnat. Lisäksi tuotantotestauksessa saadaan valmistusprosessiin liittyvää tietoa. Testauksella pyritään yleisesti mahdollisimman lähelle 100 %:n tuotevirheettömyyteen. (3.)

2.3 Testauksen luotettavuus

Testitapahtuman pitää olla luotettava ja se pitää voida toistaa koska tahansa, niin että tulos on käytännössä sama kuin edellisen. Mittaustulokseen vaikuttavat usein sähkömagneettiset häiriöt, puutteellisesti kalibroidut mittalaitteet tai väärät tai huonot testausolosuhteet. Toistettavuus tarkoittaa, että sama mittaustulos pitää tulla joka kerta samaa tuotetta mitattaessa. Kuitenkin mittaustuloksissa on aina pieni poikkeama. On mahdotonta, että eri mittauskerroilla tulokset olisivat aina täsmälleen samat. Jos näin on, voidaan olettaa, että testauslaitteisto saa saman virheellisen tuloksen joka kerta. (5, s. 18.)

2.4 Testauksen kattavuus

Alussa prototyyppien testaus on hyvin laajamittainen tapahtuma. Tuotteesta testataan kaikki sen ominaisuudet läpi eri olosuhteissa, jotta löydettäisiin tuotteen heikot kohdat. Tämä vie paljon aikaa ja taloudellisia resursseja, koska pitää tarkastaa useita tuotteita, ennen kuin tuloksia voidaan pitää luotettavana. Tästä johtuen tuotantolinjalla ei ole taloudellisesti hyvä testata jokaista tuotetta laajasti kokonaan. Tuotteesta testataan vain heikot kohdat, ja jos ne toimivat, voidaan olettaa tuotteen muidenkin osien toimivan. Tuotantotestauksessa pitää kuitenkin huomioida, ettei testausta karsita liikaa, jotta kaikki viat ja virheet varmasti löytyvät. (5, s. 11.)

Tuotantotestaus tapahtuu jo tuotantolinjalla tai välittömästi sen jälkeen. Yleensä tuotantotestaus ei sisällä toiminnallista testausta; siitä pitää huolen tuotekehityksen testauslinja. Tuotantotestaus siis keskittyy lähinnä etsimään tuotteen kokoonpanossa syntyneet viat, tai paremminkin varmistamaan tuotteen viattomuus. Mitä luotettavammin viat löydetään jo tuotantolinjalla, ennen asiakkaalle tai jatkokokoonpanoon menoa, sitä edullisemmaksi vian korjaaminen tulee. (5, s. 19.)

Tuotantotestaus voi pienissä määrissä olla manuaalista, mutta vähänkään suuremmille tuotantoerille on mielekästä toteuttaa automaattinen testaus. Tätä varten tarvitaan laitteisto, ohjelmisto ja usein myös niiden ohjaamiseen esimerkiksi PC. PC:ksi kelpaa lähes mikä tahansa riittävän tehokas tietokone. Mittalaitteisto useimmiten koostuu tavallisista, alan liikkeistä saatavissa olevista mittauslaitteistoista, esimerkiksi HP tai Agilent. Usein ohjelmisto joudutaan tekemään yrityksen sisällä, räätälöitynä kyseessä olevaan kohteeseen. (5, s. 11–12.)

Testin kattavuus on prosentuaalinen numeroarvo, joka kertoo havaittujen mahdollisten vikojen määrän suhteessa koko tuotantoon. Vikojen löytämisen helpottaminen jo suunnitteluvaiheessa (design for testability) on myös tärkeä elementti. Jos testausta ei ole otettu jo suunnitteluvaiheessa huomioon, voi olla mahdollista, ettei testauksella löydetä kaikkia vikoja, jolloin testin kattavuus jää huonommaksi. Tuotetta ei voi toimittaa asiakkaalle, ennen kuin se on riittävällä kattavuudella testattu tuotannossa syntyneiden virheiden varalta. (6, s. 19.)

2.5 Testauskustannukset

Testauksen kustannukset jakaantuvat kiinteisiin ja muuttuviin kustannuksiin. Testauksesta aiheutuvat kustannukset lisäävät suoraan tuotteen valmistuskustannuksia ja siten sen hintaa. Kiinteät kustannukset ovat kertaluontoisia, kuten testauksen suunnittelu, testilaitteiden hankinta ja käyttöönotto sekä henkilöstön koulutus laitteiden käyttöön ja testin ajamiseen. Muuttuvat kustannukset ovat jatkuvia. Näihin luetaan testaukseen tarvittavan henkilöstön palkat, testauslaitteiden ylläpito ja henkilöstön jatkokoulutus. (3.)

3 TESTAUSMENETELMÄT

Tässä luvussa esitellään pääpiirteittäin joitakin yleisimpiä elektroniikan testausmenetelmiä.

3.1 Automaattinen optinen tarkastus

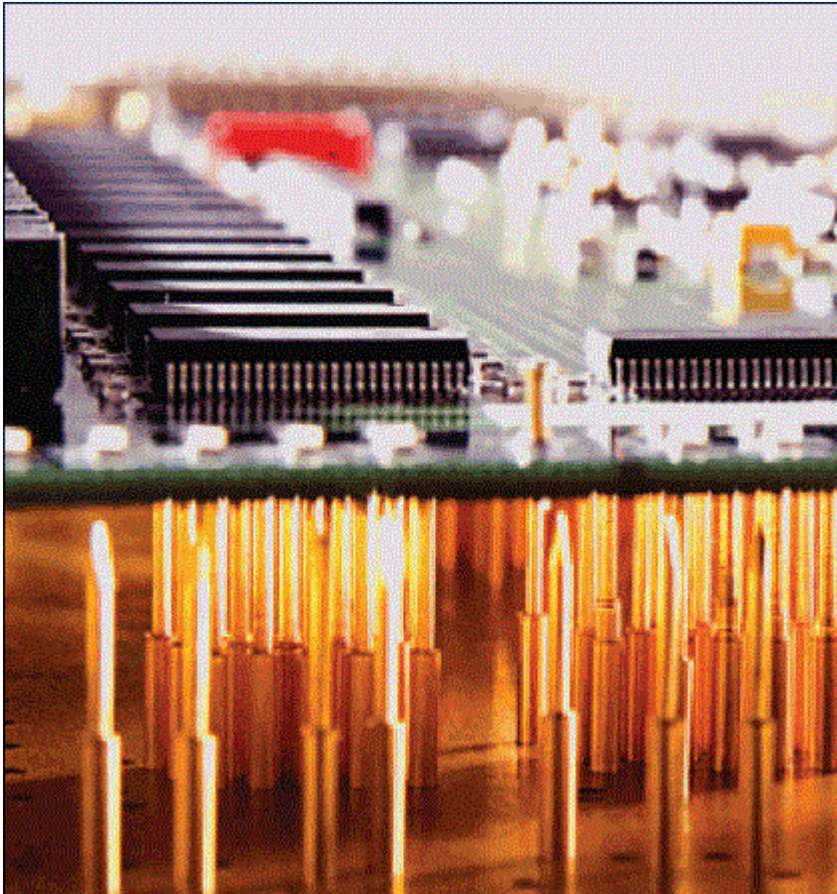
Automaattinen optinen tarkastus (AOI, Automatic Optical Inspection) sopii piirilevyn fyysisen kunnon tarkistamiseen välittömästi ladonnan jälkeen. Sillä voidaan tarkistaa juotepastan levitys piirilevyn juotosalueille sekä komponenttien ladonta piirilevylle. Ladonnan tarkistuksessa varmistetaan, että piirilevylle on ladottu oikeat komponentit oikeille kohdille oikeinpäin. Tarkastus perustuu siihen, että testikohteesta saatavaa signaalia verrataan alkuperäiseen, virheettömän tuotteen referenssikuvaan. Tämän jälkeen vertailulogiikka joko hyväksyy tai hylkää tuotteen. (3.)

3.2 Automaattinen läpivalaisutarkastus

Automaattinen läpivalaisutarkastus (AXI, Automatic X-ray Inspection) on tarkoitettu erityisesti juotosliitosten, katkosten ja oikosulkujen tarkastamiseen. Tarkastuksen hyväksyminen tapahtuu samalla tavalla kuin AOI-menetelmässä. (3.)

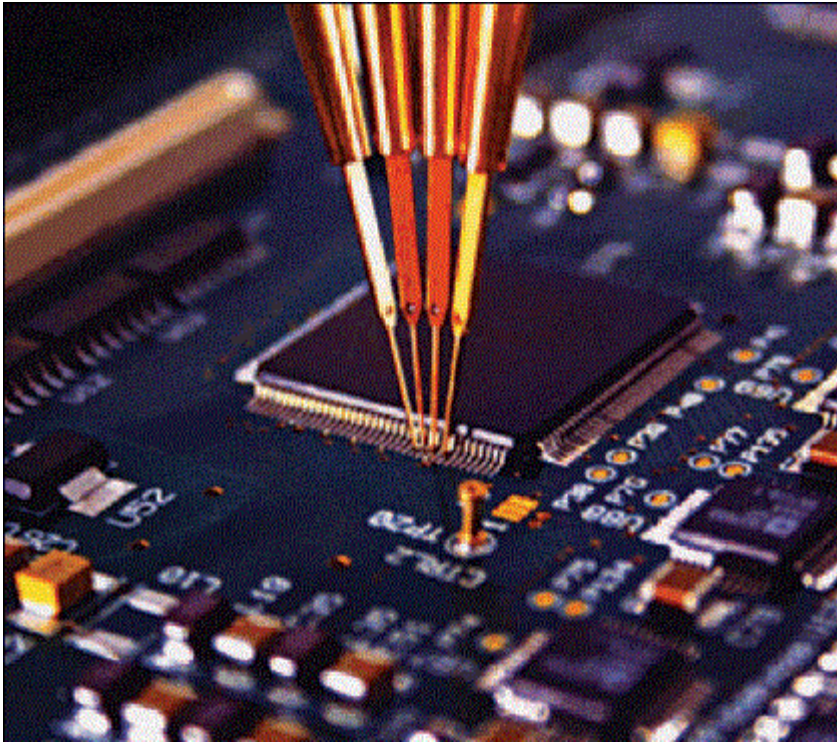
3.3 In Circuit Testing ja Flying Probe

Neulapetitestauksessa (ICT, In Circuit Testing) testauslaite eli jigi liittyy testattavaan piirilevyyn testineuloilla varustetun kytkentäyksikön avulla. Testineulat, joiden kautta yhteys luodaan, ovat kullattuja ja vaativat pääsyn kaikkiin testikohteen solmupisteisiin joko testipadien liitinnastojen tai läpivientien kautta (kuva 1). Neulapetitestauksen etuna on mahdollisuus testata laitteen komponentteja ilman käyttöjännitettä. Neulapetitestauksen haittapuolena on tarvittavien neulapetien määrä: jokainen erillinen piirilevy tarvitsee oman neulapetinsä. Neulapetitestauksessa pyritään testaamaan kaikkien passiivisten komponenttien arvot ja aktiivisten komponenttien eheys sekä avoimet juotokset ja oikosulut. (3.)



KUVA 1. Neulapetitesteri. (7.)

Flying Probe on kuin neulapetitestausta, mutta siinä ei ole kiinteää neulapetiä, vaan neuloja liikutellaan eräänlaisen käsivarren päässä haluttuihin kohtiin. Flying Probe -järjestelmässä liikkuvat neulat mittaavat etukäteen ohjelmoiduista paikoista jännitettä tai virtaa. Periaatteessa nämä toimivat lähes millä tahansa kokonsa puolesta laitteeseen mahtuville piirilevyille. Valmiit testauslaitteistot ovat kuitenkin kalliita hankintahinnaltaan, mutta toisaalta ne voidaan ohjelmoida uudelleen eri piirilevyille, jolloin jokaiselle tuotteelle ei tarvitse hankkia omaa testauslaitteistoa. Flying Probe -järjestelmä on myös kohtalaisen hidas, koska siinä on mittapää, jota täytyy joka mittauksen välissä fyysisesti siirtää. (8.) (Kuva 2.)



KUVA 2. Flying Probe -testerin mittapääät. (7.)

3.4 Boundary Scan

Boundary Scan -menetelmällä (BS) voidaan testata menetelmää tukevien piirien toiminta ja mikropiirien välisten johdotusten ja liitosten toiminta valmiilla piirilevyllä testiliittimen kautta. BS-menetelmää käytetään JTAG-väylän kautta. Vaikka JTAG onkin hidas väylä, se on kuitenkin hyvin monikäyttöinen, koska testauksen lisäksi sillä voidaan myös ohjelmoida piirilevyjä. (3.)

3.5 Toiminnallinen testaus

Toiminnallisen testauksen tavoitteena on tarkistaa, onko testattavan yksikön toiminta ja käyttäytyminen annettujen spesifikaatioiden mukainen. Toiminnallisessa testauksessa DUT:iin (Device Under Test, testattava laite) syötetään normaalia toimintaa vastaava heräte ja mitataan herätteen synnyttämä vaste. (3.)

Toiminnallisen testauksen laitteistot ovat yleensä yksilöllisiä, erillisiä testauslaitteistoja, jotka on koottu erillismittalaitteista tai mittauskorteista. Laitteisto voi-

daan joko suunnitella itse tai hankkia valmiina. Yleensä toiminnallisen testauksen laitteistossa on peruskomponentteina järjestelmä, laitteisto ja ohjelmisto.

Järjestelmä sisältää fyysisen koteloinnin, rajapinnat DUT:iin, instrumentit yhdistävät kaapeloinnit ja väylät sekä PC:n. Laitteisto taas käsittää mittalaitteiston, virtalähteet ja kuormituksen, jolla tuotetta testataan. Ohjelmisto on ohjausjärjestelmä, jolla testausta ohjataan ja suoritetaan. (3.)

3.6 Built In Self Test

Sisäänrakennetussa itsetestauksessa (BIST, Built in Self Test) tuotteeseen on sisällytetty osia, jotka synnyttävät halutut testisignaalit eli -vektorit, mittaavat niiden vasteet. Näitä mitattuja vasteita laite vertaa itse ennalta asetettuihin tavoitteellisiin tuloksiin. Mahdollisista poikkeamista tai virheellisistä tuloksista laite ilmoittaa. Muutoin laite itse hyväksyy itsensä. Koska testausosa itse sisältää mikropiirin tai järjestelmän tarvitseman testausohjelman, ei tarvetta erilliselle testauslaitteistolle ole. (3.)

BIST-toteutus kuitenkin vaatii yleensä prosessorin käyttöä. BIST-testausta on myös muokattu sen mukaan, halutaanko sillä testata koko laite vai vain sen osia, esimerkiksi mBIST, joka testaa vain muistin. (3.)

4 TESTATTAVIEN TUOTTEIDEN ESITTELY

Työssä kehitettävän testauslaitteistolla testataan Elektrobitin suunnittelemaa sotilaskäyttöön tarkoitettua VoIP-tekniikalla (Voice over Internet Protocol) toimivan kommunikointijärjestelmän laitteita. Tarkoituksena ei ole esitellä koko tuoteperheen ja yksittäisten tuotteiden toiminnallisuutta, vaan antaa kuva testattavien tuotteiden ominaisuuksista ja käyttötarkoituksista, jolloin saadaan myös kuva, millainen testauksen pitää olla.

EB Tough VoIP on sotilaskäyttöön suunniteltu VoIP-tekniikalla toimiva kommunikointijärjestelmä. Järjestelmään kuuluu muun muassa EB Network Extender, EB Tough VoIP Terminal sekä EB Tough VoIP Field Phone (kuva 3). Laitteiden kesken kommunikoidaan Push-To-Talk-tekniikalla. Käytännössä se siis toimii kuten perinteinen radiopuhelin: nappia painaessa saa yhteyden kaikkiin linjalla oleviin laitteisiin yhtä aikaa. (9.)



KUVA 3. EB Tough VoIP Network Extender (vas.), Terminal ja Field Phone (9.)

EB Tough VoIP -järjestelmä on suunniteltu toimimaan erittäin haastavissa olosuhteissa, samalla kun sen toimivuudelle on asetettu suuret odotukset, koska sotilaskäytössä ei saisi olla viallisia tuotteita lainkaan. Näin ollen kaikki virheet pitäisi löytää jo tuotannon testauksessa, eikä virheitä saa mennä läpi. (9.)

4.1 EB Network Extender

EB Network Extender on eräänlainen keskus EB Tough VoIP -järjestelmässä. Se tarjoaa VoIP-terminaalin, Ethernet-kytkimen sekä monikanavaisen SHDSL-

modeemin. Sillä saadaan luotettava langallinen yhteys ja VoIP-palvelut kentälle. Laajemman verkon saa liittämällä useita EB Network Extendereitä ketjuun, jopa kymmenen kilometrin väleillä. Yksikköön voidaan kiinnittää kaiutin tai samat kuulokkeet kuin terminaaliin (9). EB Network Extenderin toiminnallinen piirilevy on nimeltään MSU, jota nimitystä piirilevystä tässä työssä käytetään.

4.2 EB Tough VoIP Terminal

EB Tough VoIP Terminal on kannettava päätelaite. Terminaalin voi kytkeä Extenderiin tai muuhun saman järjestelmän laitteeseen. Terminal toimii PTT (Push-To-Talk) -tekniikalla ja tukee PoE:ä (Power over Ethernet), eli se saa käyttöjännitteensä datalinjalta Ethernet-väylästä. (9). EB Tough VoIP Terminalin toiminnallinen piirilevy on nimeltään MCU.

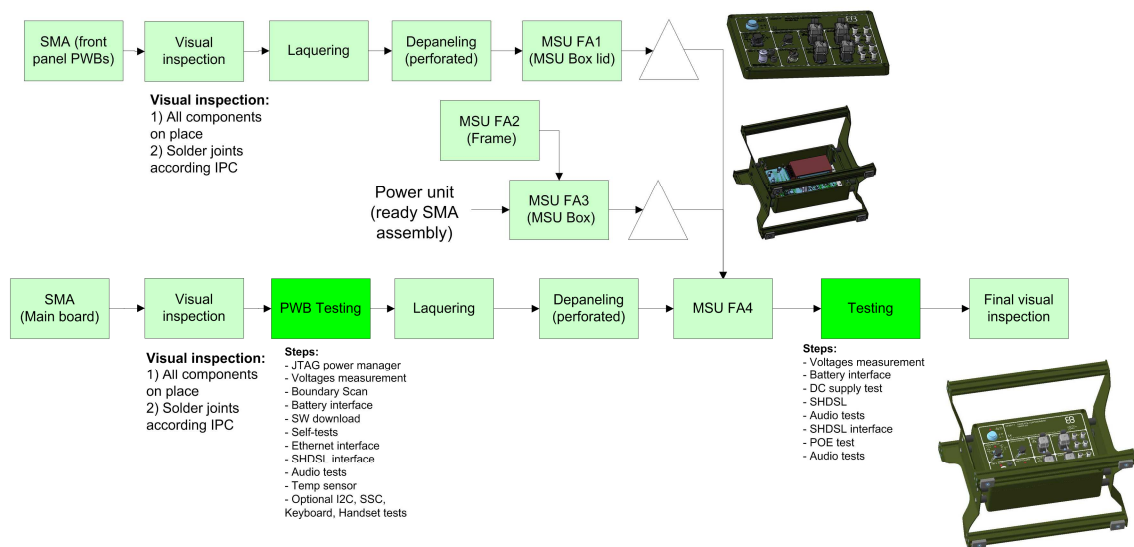
4.3 EB Tough VoIP Field Phone

EB Tough VoIP -tuoteperheeseen kuuluu lisäksi olennaisena osana Field Phone, joka kuitenkin rajautuu ulos tämän työn tavoitteista. Field Phonen piirilevynä on MFU, jota tullaan tulevaisuudessa testaamaan samalla testerillä. (9.)

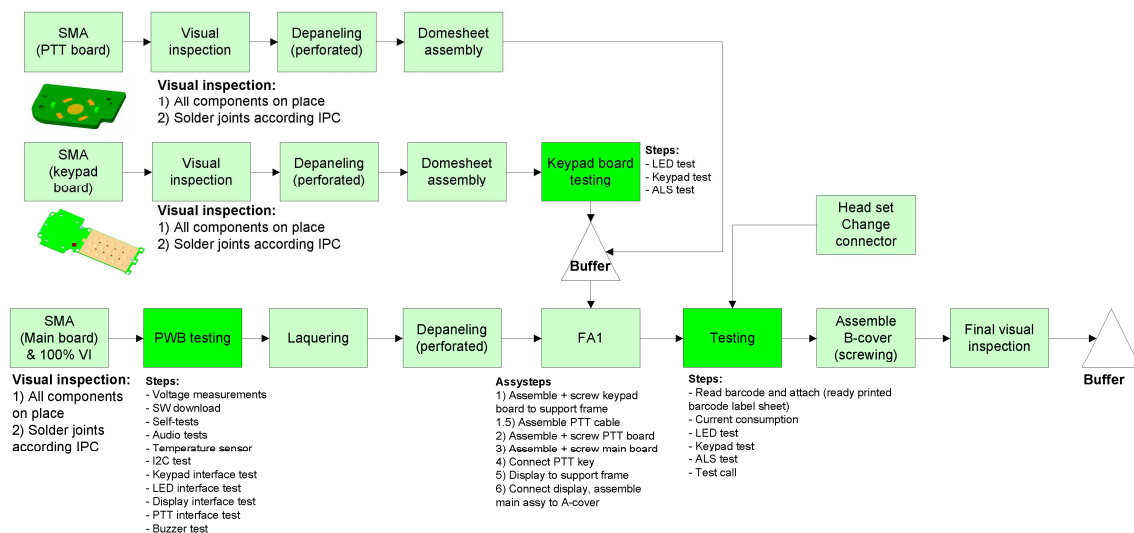
5 TESTIJÄRJESTELMÄN NYKYTILA

Tässä luvussa esitellään testitornin ja siihen liittyvien laitteiden ja väylien nykytilaa eli sitä tilaa, jota on tarkoitus optimoida.

Testerit sisältää vain tietyt toiminnalliset testaukset. Työssä keskitytään vain kuvissa 4 ja 5 esitettyjen tuotantokaavioiden PWB Testing -vaiheeseen. Muissakin vaiheissa tuotantoa luonnollisesti on testausta, mutta tämä työ on tarkoituksella rajattu tähän vaiheeseen.



KUVA 4. MSU-piirilevyn tuotantokaavio (liite 3, s. 7)

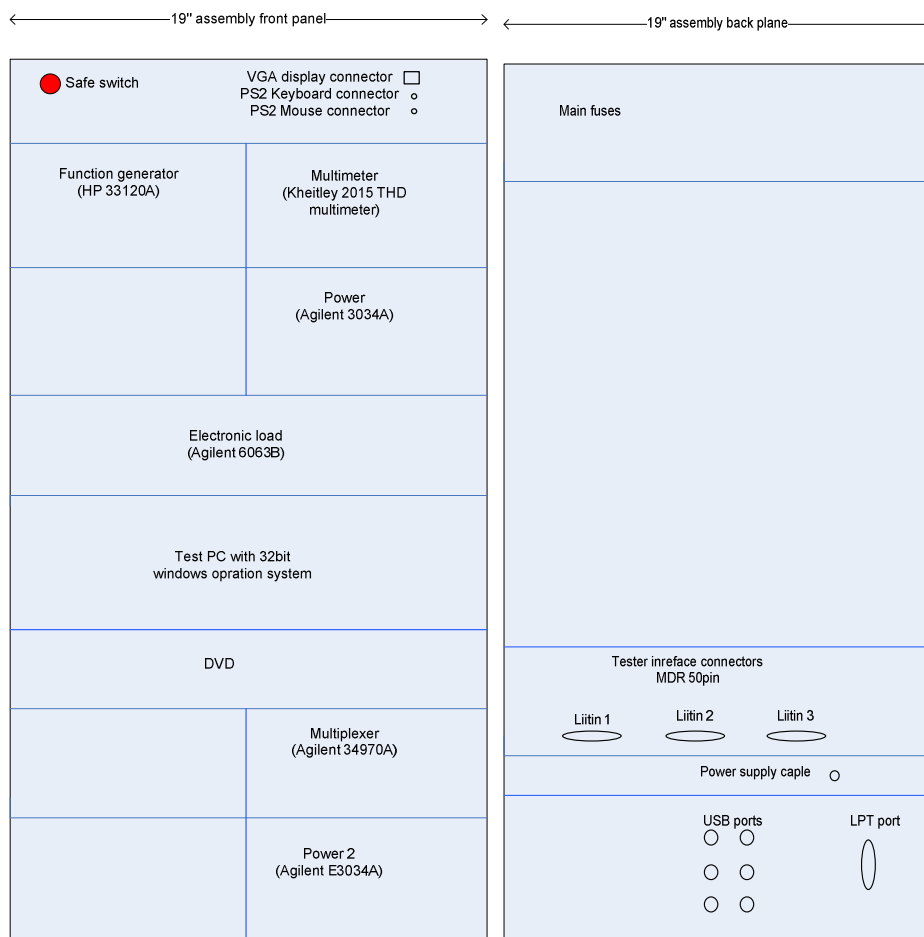


KUVA 5. MCU-piirilevyn tuotantokaavio (liite 4, s. 7)

5.1 Testitorni

Alkuperäinen testitorni koostui yleisesti saatavilla olevista mittalaitteista. Järjestelmässä ei ollut mitään ylimääräistä, vaan se oli suunniteltu ja toteutettu nimenomaan tähän järjestelmään. Testerillä pitäisi saada testattua sekä MCU-että MSU-piirilevyt sekä lisäksi mahdollisesti MFU-piirilevy.

Torni on tavallinen, standardikokoinen metallinen siirrettävä räkki. Tornissa on kiinni testi-PC 32-bittisellä Microsoft Windows XP Pro -käyttöjärjestelmällä, yleismittari (Keithley 2015 THD multimeter), funktiogeneraattori (HP 33120A), elektroninen kuorma (Agilent 6063B electronic load), multiplekseri (Agilent 34970A) sekä kaksi virtalähdettä (Agilent 3034A) (kuva 6).



KUVA 6. Testitornin etupaneeli ja takapaneeli (liite 2, s. 7–8)

5.2 Jigit

Jigeinä toimivat nimenomaan MCU:n ja MSU:n piirilevyille suunnitellut testialustat, kummallekin omansa. MCU:n jigi on eräänlainen neulapeti, jonka neulat koskettavat tiettyjä pisteitä piirilevyllä (liite 4, s. 15). MSU:n jigi taas koostuu kolmesta liittimestä ja testaus tapahtuu BS-menetelmällä. (Kuva 7.)



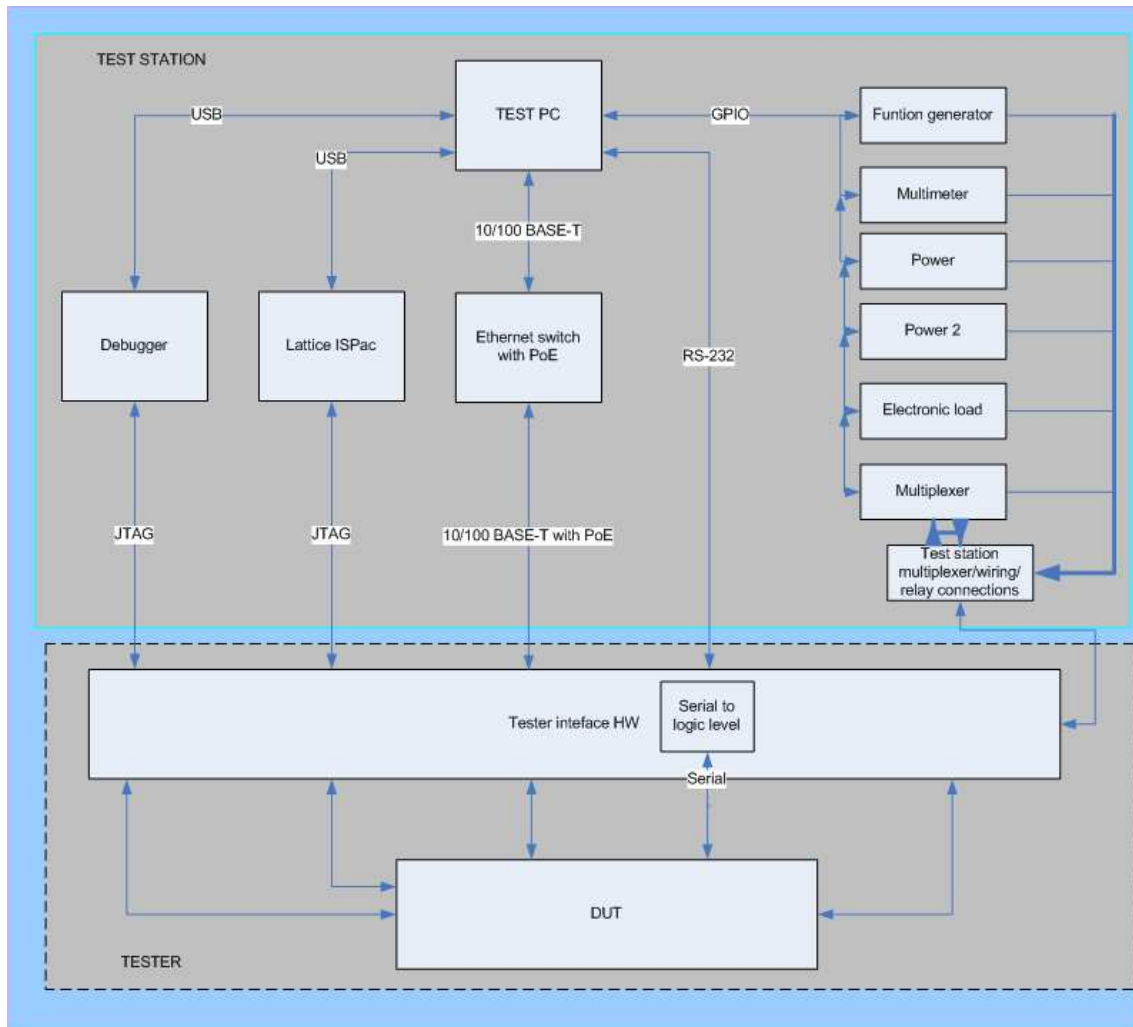
KUVA 7. MSU:n testijigi ja ohjelmointikortti

MSU:n jigi on hiukan kömpelö. Sen käyttämisessä on monta vaihetta. Aluksi kaikki kolme liittintä pitää kiinnittää siihen. Sen jälkeen testattava kortti asetetaan paikalleen ja sen päälle tulee vielä ohjelmointikortti. Ohjelmointikorttiin kiinnitetään Ethernet- ja JTAG-johdot, jos niitä ei vielä ole kiinnitetty. Tähän kaikkeen kuluu monta minuuttia.

MCU:n jigin suunnitteluun on selkeästi käytetty enemmän aikaa. MCU-kortti kiinnitetään tukevilla kytkimillä tiukasti vasten neulapedin neuloja. Lisäksi

MCU:n testaaminen vaatii vain liittimen 2 käyttöä, eikä lainkaan erillistä ohjelmointikorttia.

Kuvassa 8 näkyvät testitornin yhteydet sekä mittalaitteiden käyttämät väylät. Kuvasta näkee, että testi-PC on yhteydessä DUT:hen kolmella eri tavalla: JTAG-debuggerin kautta, Lattice ISPac:n kautta sekä Ethernetillä. Tämän lisäksi vielä mittalaitteet vaativat oman yhteytensä.



KUVA 8. Kaavio testerin yhteyksistä (liite 2, s. 5)

5.3 Liitännät ja väylät

Järjestelmä on koottu lukuisia eri väyliä käyttäen. Tietokone kommunikoi mittalaitteiden kanssa GPIB-väylän kautta testijgin kanssa liittimien 1, 2 ja 3 kautta. Lisäksi alustavan ohjelman lataus testattavalle kortille tapahtuu JTAGin kautta

ja varsinaisen ohjelmiston lataus Ethernetin kautta. JTAGin käyttö vaatii myös JTAG-muuntimen. Tämän virkaa hoitaa Lauterbachin toimittama JTAG-debugger, jonka luotettavuus on tässä tapauksessa käytön aikana osoittautunut huonoksi.

5.4 Ohjelmisto

Järjestelmä toimii 32-bittisen Windows XP Pron päällä. Itse testiohjelmisto on toteutettu TTCN-3-skriptejä käyttäen OpenTTCN-ohjelmistolla. Skriptit ohjaavat niin mittalaitteita kuin testattavaa korttia ja sen jigiä. Samalla ne luovat automaattisesti lokia jokaisesta toiminnostaan. Tätä lokia voi katsella sekä testin aikana että sen jälkeen.

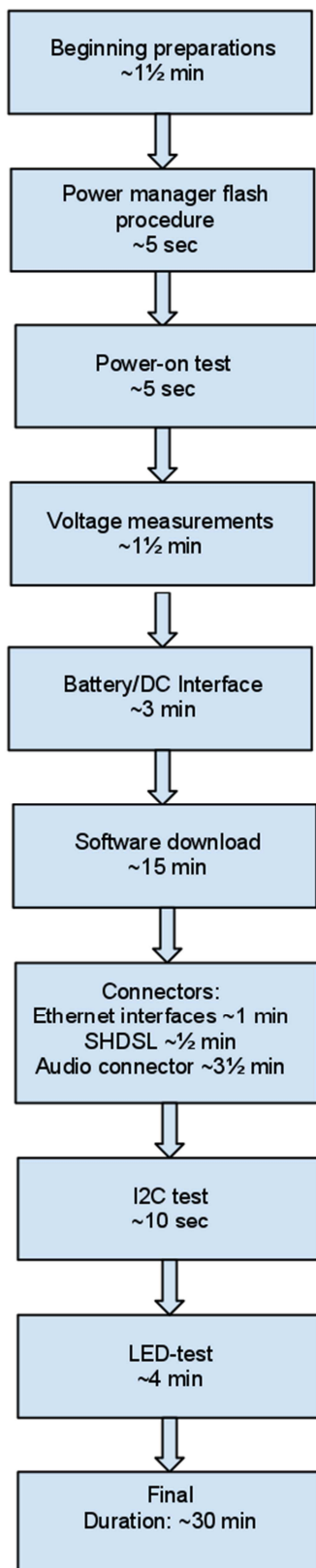
Tietokoneella on myös National Instrumentsin Measurement & Automation -ohjelmisto, jolla voi manuaalisesti tarkastaa mittalaitteiden ja väylien tilaa.

5.5 Testisekvenssin rakenne

Kokonaisuudessaan testausjärjestelmä suorittaa kaikki eri testit lineaarisesti peräkkäin. Näiden järjestys on tärkeä: testeissä löydettyjen mahdollisten virheiden prioriteetti on määritettävä. Toisaalta on myös otettava huomioon, onko edes mahdollista testata jotain ennen jotain toista osa-aluetta. Tällä hetkellä osa testeistä vaatii, että piirilevyllä on ajettu ohjelmisto, osa testeistä taas täytyy ajaa ilman ohjelmistoa.

5.5.1 MSU

MSU:n testaus etenee kuvan 9 mukaisesti.



Kuva 9. MSU:n testin eteneminen

Aluksi tehdään alustavia valmisteluja. Näihin kuuluu mittalaitteiden ja testi-PC:n käynnistäminen, piirilevyn kiinnittäminen testijigiin, tarvittavien kaapeleiden ja väylien kiinnittäminen sekä sarjanumeron luku tietokoneelle viivakoodinlukijalla piirilevyn päällä olevasta tarrasta.

Ensimmäisenä testinä MSU:lle on Power Manager -testi, jossa ajetaan alustava testiohjelmisto levyille. Tämä testiohjelmisto sisältää tuen tuleville testeille ja ohjelmiston lataukselle. Power Manager ohjelmoidaan JTAG-väylän kautta.

Tämän jälkeen, testin toisessa vaiheessa, suoritetaan levyille Power-on test, jolla varmistetaan levyn käynnistyminen. Tämä testaa laitteen virtakytkimen toiminnan kytkemällä aluksi kytkimen maahan. Tällöin jännitteen pitäisi olla 0 V 3,3V_SHDSL-linjalla. Tämän jälkeen piiri jätetään avoimeksi.

Kolmantena testinä on Voltage measurements eli mitataan tiettyjen pisteiden väliset jännitteet ilman ohjelmistoa.

Jännitemittausten jälkeen, viidentenä, tulee akun ja DC-jännitteen toiminnan testaus eli Battery/DC interface tests. Käytännössä se tarkoittaa sitä, että akun jänniteindikaattoreina olevat ledit toimivat oikein. Lisäksi testataan virrankulutus sekä 12 että 28 voltilla. Viimeisessä vaiheessa testataan virrankulutuksen rajoitin.

Kuudentena levyille ladataan varsinainen ohjelmisto. Aluksi ladataan U-bootia käyttäen JTAG-väylällä alustava ohjelmisto, jolla saadaan käyttöön muut liitännät sekä saadaan kirjoitettua levyille tärkeimmät tiedot, kuten sarjanumero ja MAC-osoite. U-Boot tallentuu laitteen SDRAM-muistiin (Synchronous Dynamic Random Access Memory). Suoritin lataa sen jälkeen U-Bootin muistista ja sen jälkeen varsinaisen ohjelmiston Ethernet-väylää pitkin. Tämä varsinainen ohjelmisto sisältää laitteen toiminnallisuuden. Tähän tarvitaan joko Linux- tai Windows-käyttöjärjestelmällä toimiva TFTP-palvelin (File Transfer Protocol).

Seitsemäntenä on Ethernet-liitännän testaus. Tämä tapahtuu kytkemällä viisiporttinen Ethernet-kytkin MSU:hun. Kahdeksas vaihe on SHDSL-linjan testaus. Tässä testataan SHDSL-liitinten (Single-pair high-speed digital subscriber line) toiminta tekemällä takaisinkytkentä SHDSL 1- ja SHDSL 2 -porttien välille. Tä-

män jälkeen toisesta lähetetään dataa ja toisessa tarkistetaan datan perilletulo ja oikeellisuus.

Yhdeksäntenä testataan audioliitettä. Sitä varten kytketään DSP:stä päälle takaisinkytkentätoiminnallisuus MIC1-HOP1-välille. Tämän jälkeen takaisinkytkentä testataan viidellä eri taajuudella. Sen jälkeen testataan vielä MIC1-LSP-takaisinkytkentä. Viimeisenä vaiheena audiotestissä on CODE- ja JOKE-signaalien toiminta. Tämä tapahtuu kytkemällä ne maahan, ja tulos voidaan lukea prosessorilta.

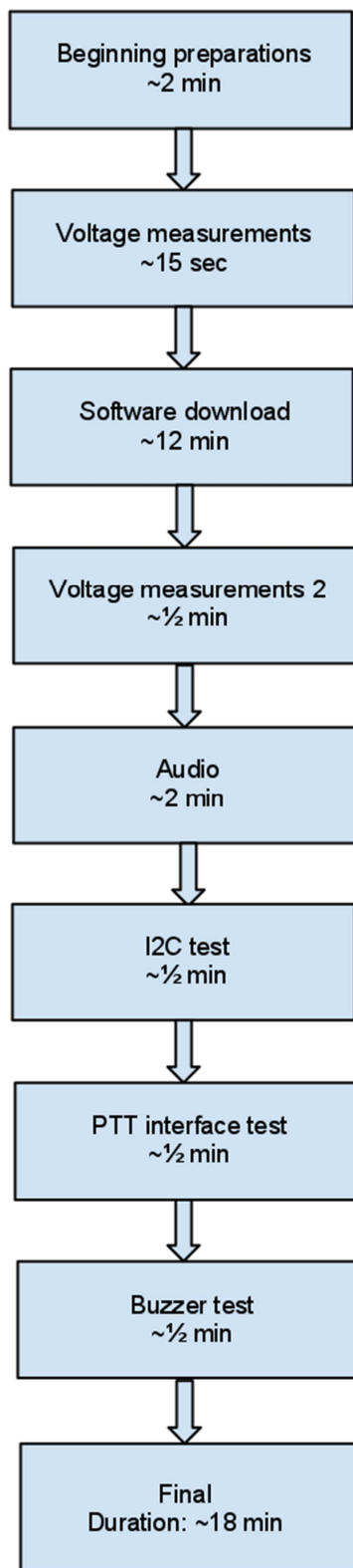
Kymmenentenä kokeillaan I²C-väylän toimivuus. Tässä testissä I²C-väylä testataan lukemalla lämpötila-anturin arvo ja vertaamalla sitä rajoihin.

Yhdestoista ja viimeinen on LED-testi. Tässä testissä sytytetään vuorotellen erivärisiä ledejä. Testerin käyttäjän pitää kuitata eri ledien syttyminen, ennen kuin testi pääsee etenemään.

Testi vie kokonaisuudessaan aikaa noin 30 minuuttia. Tarkempi ajan jakautuminen näkyy kuvassa 9. Eniten aikaa vie ohjelmiston lataus, joka haukkaa noin puolet testiajasta. Sen lisäksi ledien testaus vie myös aikaa, sekä akun jänniteindikaattoritestissä että loppuvaiheen led-testissä.

5.5.2 MCU

MCU:n testaus etenee kuvan 10 kaavion mukaisesti.



KUVA 10. MCU:n testin eteneminen

Ensimmäiseksi laite kiinnitetään testijigiin. Tämän jälkeen siihen kiinnitetään tarvittavat kaapelit, kuten I²C. Piirilevy lukittuu siististi paikalleen, kun se on oikein kiinnitetty. Aluksi laitteelta testataan jännitemittaukset ilman ohjelmistoa. Tässä vaiheessa siis piirilevylle laitetaan käyttöjännite päälle ja mitataan tietyistä pisteistä mittaukset. Näiden testipisteiden sijainti näkyy liitteessä 4 sivulla 15.

Ensimmäisten jännitemittausten jälkeen levyille ladataan ohjelmisto. Aluksi ladataan U-bootilla JTAG-väylän alustava ohjelmisto, jonka avulla voidaan ladata varsinainen ohjelmisto Ethernet-väylää pitkin. JTAG vaatii testitorniin Lauterbach-moduulin, joka on kiinni USB-väylässä. Tämä U-boot-lataus toistetaan tuotannon aiemmin vaatineista syistä 4–5 kertaa testausta kohden. Ohjelmiston lataus levyille kestää noin puolet koko testauksen ajasta.

Ohjelmiston latauksen jälkeen tehdään lisää jännitemittauksia. Nämä mittaukset voidaan nykymuodossaan tehdä vasta, kun ohjelmisto on onnistuneesti ladattu piirilevylle.

Neljäs vaihe testissä on audiotestit. Näissä MIP1-HOP1-loopback-testeissä DSP asetetaan loopback-tilaan. Tämä tarkoittaa lähetetyn signaalin ohjaamista takaisin lähtöpisteeseen muokkaamatta sitä. Tämän jälkeen mittaus suoritetaan viidellä eri taajuudella, kustakin mitataan jännitteet ja niitä verrataan raja-arvoihin.

Viidentenä vaiheena on I²C-väylän testit. Testaus tapahtuu lukemalla AD-muuntimen jännite I²C-väylän kautta. Jos jännitteen luku onnistuu, voidaan todeta I²C-väylän olevan kunnossa.

Kuudentena seuraa PTT-testit. Niissä testataan Push-To-Talk-napin eli tangentin painamisen toiminta: merkkiledin pitää syttyä ja tietyt signaalit pitää havaita, jotta voidaan todeta, avautuuko linja.

Seitsemäs ja viimeinen testi on äänimerkin eli buzzerin testaus. Tämä vaatii työntekijän läsnäolon, koska testissä kirjaimellisesti annetaan äänimerkki, jonka kuittausta ohjelma jää odottamaan käyttäjältä.

Koko testi vie aikaa kokonaisuudessaan noin 18 minuuttia. Tästä valtaosa eli noin 12 minuuttia kuluu ohjelmiston lataukseen.

6 TESTERIN TAVOITETILA

Luvussa 5 näkyy testerin nykytila, joka ei ole oikein hyvin optimoitu massatuotantoa ajatellen. Tavoitteena on siis lyhentää piirilevyn testaukseen kuluva aikaa oleellisesti, mielellään neljäsosaan alkuperäisestä. Samaan aikaan pitää pystyä varmistumaan koko levyn täydellisestä toiminnasta, sillä tuotteen käyttökohteesta johtuen tuotantotesteristä ei saa mennä läpi viallisia levyjä.

Testin kesto siis pyritään lyhentämään noin viiteen minuuttiin pienillä mutta oleellisilla muutoksilla. Kaikki muutokset pyritään toteuttamaan olemassa olevalla laitteistolla, mutta siihen voidaan tehdä pieniä muutoksia.

6.1 Fyysiset ja mekaaniset muutokset

Nykytilassaan testerin hitain osuus on ohjelmiston lataus. Osa tästä heijastuu myös alkuvalmisteluihin: MSU:n testissä pitää kiinnittää ohjelmointilevy ja sille kaapeloinnit. JTAG-liitäntä on itsessään helppo käyttää, mutta sen käyttöönotto kokonaisuuden kannalta on vaikeaa: se vaatii väliin debuggerin, joka taas on hyvin epäluotettava.

Ohjelman latauksen aluksi ladataan siis I²C-väylän kautta pieni image, joka sisältää tuen varsinaisen ohjelmiston lataukselle. Olemassa olevaa imagea muutetaan siten, että se sisältää tuen vain vaadittaville testeille. Varsinainen ohjelmiston lataus siis jätetään tästä vaiheesta pois. Tällä hetkellä piirilevyillä oleva Power Manager vaatii alussa ohjelmoinnin JTAG-väylän kautta. Tämä voidaan ratkaista esimerkiksi esiohjelmoinnilla, jossa piirille ladataan sen ohjelmisto jo piirin tuotantolinjalla. Näitä esiohjelmoituja piirejä on tosin jälkikäteen vaikea ohjelmoida uudelleen esimerkiksi päivityksen tai huollon yhteydessä. Toisaalta voi olla mahdollista vaihtaa Power Manager -piiri yhteensopivammaksi. Se tosin vaatii suuremmat toimenpiteet koko tuotesarjan kannalta.

Kun alustava, testeille tuen sisältävä ohjelmisto ladataankin I²C:n kautta, voidaan erillinen I²C-testi jättää kokonaan väliin, koska I²C tulee testatuksi ohjelmiston latauksen yhteydessä. JTAG-väylälle sen sijaan pitää mahdollisesti kehittää jokin pieni testi, jossa senkin toimivuus voidaan varmistaa.

On myös mahdollista, että I²C:n tai JTAG:n kautta ladataan alussa image eli levynkuva, joka sisältää tuen vain Ethernet-väylän käyttöönololle. Ethernet-väylä on ylivoimaisesti nopein näissä laitteissa olevista liitännöistä, joten olisi perusteltua ottaa se käyttöön jo varhaisessa vaiheessa.

MSU:n testijigi toimii kolmella liittimellä: Liitin 1, Liitin 2 ja Liitin 3. Kun jotain halutaan mitata, näiden kautta annetaan syöte. Tämän jälkeen joudutaan odottamaan vastetta. Jokaiselle signaalille on varattu liittimistä oma pinni ja johdin; samaa johdinta ei käytetä kahteen kertaan. Tämä aiheuttaa käytännön toteutuksessa runsaasti eri johtimia, jolloin testerin ylläpito on selkeästi hankalampaa. Nopeampi keino on hoitaa sama asia neulapedillä, kuten MCU-testissä. Tällöin haluttu jännite- tai virtamittaus voidaan suorittaa suoraan halutuista pisteistä, eikä tarvitse jäädä odottamaan herätettä tai vastetta. MSU:n jigi on muutoinkin hiukan kömpelö ja epäkäytännöllinen.

Lopussa oleva Buzzer-testi voidaan myös automatisoida siten, että laitteistoon lisätään pieni mikrofoni, joka reagoi läheltä tulevaan, tarpeeksi kovaääniseen ärsytykseen. Tätä mikrofonia ohjataan tietokoneelta automaattisesti, samoin kuin muutakin kokoonpanoa. Tällä tavoin voidaan määrittää äänimerkki antamaan ääntä vaikka vain kaksi sekuntia, jonka aikana mikrofonilla tutkitaan, tuleeeko ääntä. Tämän vaiheen automatisointi poistaa viimeisemmätkin laitteiston käyttäjästä johtuvat viiveet, ja saadaan taas säästettyä jonkin verran aikaa, koska ei voida olettaa, että operaattori on koko ajan valmiudessa sormi napilla valmiina kuittaamaan äänen.

6.2 Ohjelmalliset muutokset

6.2.1 Testerin ohjelmointiympäristö

Testerin koodi on toteutettu TTCN-3-skripteillä. TTCN-3 on nimenomaan testeympäristöön kehitetty skriptikieli, mutta se ei ole kovin loppukäyttäjäystävällinen eikä havainnollinen. Ympäristön ohjelmoija on perillä testerin toiminnasta, mutta testerin varsinainen käyttäjä ei useinkaan ole ohjelmoija. Tästä johtuen ympäristö kehitetään uusiksi. Sama laitteisto säilytetään mutta ohjelmointiympäristöksi vaihdetaan National Instrumentsin LabView. LabView on havainnollinen

sekä käyttäjälle että ohjelmoijalle. Sen graafinen ympäristö näyttää selkeästi ongelmakohdat, ja vikatilanteissa se antaa selkeän virheilmoituksen ja vikakoodin, jonka voi antaa ohjelmoijalle. Näin vikatilanteet saadaan selvitettyä huomattavasti helpommin ja nopeammin, samalla kun testerin ylläpidettävyys paranee.

LabView'ssa on myös selkeästi helpommin toteutettu mittalaitteiden hallinta ja ajurit. Nämä löytyvät selkeinä kuvakkeina lähes kaikille markkinoilla oleville mittalaitteille, ja niille joille ei löydy, ne on helppo luoda itse. LabView sisältää myös toiminnallisuudet mittaustulosten siirtämiseksi mihin tahansa tiedostoon, esimerkiksi taulukkolaskentaan. Tämä helpottaisi myös mittaustulosten säilytystä, luettavuutta ja myöhempää käsittelyä; nykyiset TTCN-3 -skriptit kun tuottavat vain tavallisen tekstitiedoston.

Jos ympäristön kieli on mahdoton muuttaa LabView'ksi, vaikka tuotannollisista syistä (samaa testeriä pitäisi muuttaa liian monessa paikassa), pitää TTCN-3 – skripti optimoida. Tällä hetkellä esimerkiksi U-boot ajaa monta kertaa peräkkäin saman, alustavan ohjelman piirilevyn SDRAM-muistiin. Periaatteessa laitteen toiminnalle tästä ei ole haittaa, vaan uusi vain kirjoittuu vanhan päälle, mihin tuhlautuu monta minuuttia. LabView-kehitysympäristöstä löytyy tuki myös Linux-käyttöjärjestelmälle. (10.)

6.2.2 Testerin käyttöjärjestelmä

Koska testattavat laitteet ovat Linux-pohjaisia, olisi myös testerin hyvä olla Linux-pohjainen. Tällöin monet ajuri- ja yhteensopivuusongelmat poistuisivat. TTCN-3 toimii yhtä hyvin Linux-pohjaisissa PC:ssä kuin myös Windows-pohjaisissa (11). Lisäksi kumpaankin voidaan käyttää samaa OpenTTCN-ohjelmistoa, jolloin alustassa käytettävän käyttöjärjestelmän muutos ei aiheuttaisi itse testerille mitään ongelmaa.

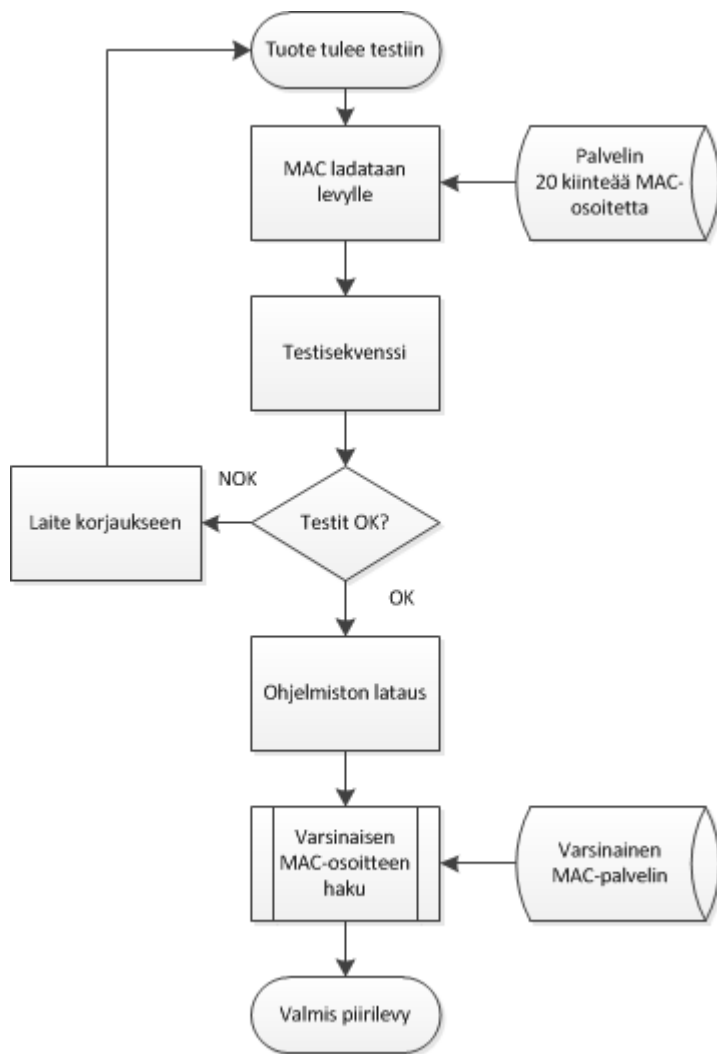
Käyttöjärjestelmän vaihto Windows-ympäristöstä Linux-ympäristöön tuo myös luotettavuusparannuksen. Linux-ympäristö pitää kertaalleen laittoa kuntoon kaikkine parametreineen, jonka jälkeen siitä voidaan luoda eräänlainen valmis image eli levynkuva. Jos ja kun järjestelmässä oleva tietokone pettää, voidaan se korvata millä tahansa tietokoneella, ladata tämä alussa tehty järjestelmäima-

ge tietokoneeseen ja jatkaa testausta lähes häiriöttä. Mitään ajuri-, ohjelmisto- tai muitakaan asennuksia ei tässä vaiheessa enää tarvitse tehdä. Tämä parantaa testerin käyttöönoton nopeutta myös mahdollisessa ongelmatilanteessa.

6.2.3 MAC-osoite

MAC-osoite (Media Access Control) on yksilöllinen laitteistotason osoite verkkokortilla. Jokaisessa verkkokortissa on siis oma MAC-osoitteensa. Osoite koostuu kuudesta kaksinumeroisesta heksadesimaaliluvusta. Näistä kolme ensimmäistä paria ovat valmistajan itselleen varaamia ja kolme jälkimmäistä paria ovat juoksevaa numerointia. Tällä hetkellä MAC-osoite haetaan tekstitiedostosta, josta tarkistetaan edellinen osoite, lisätään siihen 1 ja syötetään sen jälkeen kortille. Tämä on hyvin epäkäytännöllinen ja epäluotettava tapa, koska mahdollisen tietokatkoksen tullessa saattaa sama MAC-osoite mennä kahteen eri laitteeseen tai toisaalta jokin MAC-osoite jää käyttämättä. MAC-osoitteen haku siis korvataan automaattisella ohjelmalla, joka tarkistaa MAC-osoitteen saatavuuden. Tätä varten yhtiöllä on jo olemassa oma sovellutuksensa, jossa MAC-osoitteet sijaitsevat yhtiön palvelimilla. Testerin tarvitsee vain hakea sieltä tarvitsemansa määrä MAC-osoitteita omaan käyttöönsä. Tätä ei vain ole vielä otettu käyttöön tässä työssä esiteltujen tuotteiden testeissä.

MAC-osoite on pakko ladata jo testisekvenssin alkuvaiheessa, koska Ethernet-väylän käyttö edellyttää laitteelta MAC-osoitetta. Tästä johtuen on vaara, että MAC-osoitteita menee viallisten tuotteiden mukana hukkaan. Jos tämä hukkaan menevä MAC-osoite otetaan uusiokäyttöön, on vaara, että sama MAC-osoite on kahdessa paikassa yhtä aikaa. Tämä voidaan ohittaa osoittamalla tuotantotesterille esimerkiksi 20 eri MAC-osoitetta siten, että testauksessa olevilla laitteilla käytetään vain näitä peräkkäin. Lopullinen MAC-osoite ladataan kortille vasta, kun voidaan olla varma, että laite toimii, eli esimerkiksi lopussa olevan ohjelmiston latauksen yhteydessä. Tällöin MAC-osoitteita ei mene hukkaan, eikä ole vaaraa myöskään siitä, että markkinoille lähtee laitteita, joilla on sama MAC-osoite. Kuvassa 11 on esitetty MAC-osoitteen haku lohkokaaaviona.



KUVA 11. MAC-osoitteen haku

6.2.4 Mittaustulosten käsittely

Mittaustulokset tallentuvat nykymuodossaan tekstitiedostoon (liite 5). Testerin tuottama yhteenveto testistä on selkeä ja informatiivinen, mutta sen muoto tekstidokumenttina ei ole käytettävyydeltään hyvä. Se korvataankin taulukkolaskentatiedostolla, jossa on testin jälkeen helppo etsiä ja jaotella tietoja. Samoin tulosten jälkikäsittely helpottuu huomattavasti. Tuloksia ei ole niin paljon, että niistä kannattaisi varsinaista tietokantaa tehdä.

6.3 Testin rakenteen muutokset

Koko testisekvenssin rakennetta muutetaan siten, että alussa piirilevyille ladataan ohjelmisto, joka sisältää tuen testeille sekä MAC-osoitteen. Tällöin tarvitta-

vat testit voidaan ajaa, mutta ei tarvitse odottaa varsinaisen ohjelmiston latautumista keskellä testausvaihetta.

Alun perin ohjelman lataus oli keskellä testausta. Tämä aiheutti testaukseen erittäin pahan pullonkaulan: ohjelmiston lataus vei yli puolet testausajasta keskellä testiaikaa. Nyt kun alussa ladataan vain tuen testeille sisältävä ohjelmisto, joka on kooltaan erittäin paljon pienempi kuin varsinainen ohjelmisto, säästetään aikaa sen latauksessa. Lisäksi kun tehdään kokonaan toinen, erillinen yksikkö, jossa tapahtuu vain ohjelmiston latausta, säästetään aikaa vielä lisää, koska näin saadaan tehtyä manuaalisesti moniajaoja. Samaan aikaan kun laitetta A testataan, laitteeseen B, joka on jo testattu, voidaan ladata ohjelmistoa. Koska joka tapauksessa kyseessä oleva testitorni tarvitsee operaattorin, sama operaattori voi suorittaa samaan aikaan myös ohjelmiston latauksen. Ohjelman lataus siis siirretään kokonaan omaan testitorniinsa. Tähän torniin tarvitaan teholähde piirilevylle, Ethernet-liitäntä sekä PC näiden ohjaukseen.

Koska ohjelmiston lataus kestää joka tapauksessa noin 15 minuuttia, kannattaa ohjelmistoa ladatessa muutoinkin käyttää moniajaoja. Niinpä ohjelmointipisteeseen pyritään saamaan yhtä aikaa vähintään kolme eri laitetta, jolloin sinä aikana, kun kolmeen laitteeseen ladataan ohjelmistot, saadaan varsinaisella testerillä testattua kolme seuraavaa levyä.

6.4 Vaihtoehtoiset testausmenetelmät

Piirilevyn testaukseen on useita eri tekniikoita, joita esiteltiin tämän työn alussa luvussa 3. Näistä valittiin pari, jotka olivat jo ennestään kiinnostaneet Elektrobiitiä, ja selvitettiin hiukan niiden sopivuutta tähän järjestelmään. Varsinaisia mittauksia ei voitu tehdä, vaan kaikki esitetyt asiat ovat arvioita testimenetelmän sopivuudesta tähän järjestelmään ja näille piirilevyille.

Flying Probe on suosittu elektroniikan automaattinen testauslaite. Laite koostuu kahdesta tai useammasta eräänlaisen käsivarren päässä olevasta, liikkuvasta ja kallisteleavasta mittauspäästä. Piirilevy voi pysyä paikoillaan tai olla sekin liikkuvalla alustalla. Laite ohjelmoidaan siirtämään mittauspää tiettuihin pisteisiin piirilevyllä. Itse mittaukset eivät kestä kovin kauan, mutta varsinkin jos mittapis-

teitä on paljon, aikaa kuluu mittauspäiden liikuttamiseen. Tässä työssä mainittujen tuotteiden tuotantoerät eivät kuitenkaan ole niin isot, että sillä olisi kovin suurta merkitystä.

Flying Probella on hyvä testata erityisesti passiiviset komponentit: vastukset, kelat ja kondensaattorit. FP-testeri kannattaisikin sijoittaa välittömästi näiden komponenttien ladonnan jälkeen. Näin saataisiin suurin hyöty Flying Probesta: jos passiivisissa komponenteissa on vikoja, voidaan kyseinen viallinen levy poistaa linjalta jo ennen aktiivisten komponenttien ladontaa.

Flying Probenkin voisi korvata jo ladontavaiheessa latojalla, joka osaa jo latoessaan mitata esimerkiksi vastusten ja muiden passiivisten komponenttien sähköiset arvot. Mikäli sähköinen arvo poikkeaisi määrätystä, kone hylkäisi komponentin ja hakisi automaattisesti tilalle uuden. Mikäli tämä toistuu useamman kerran, kone pysähtyy ja ilmoittaa virheellisestä komponenttirullasta. Tällä saataisiin inhimilliset virheet poistettua jo mahdollisimman aikaisin: kone huomaa heti latoessaan, että sen sisälle on asetettu virheelliset komponentit. (12.)

6.5 Testikattavuus

Nykymuodossaan testeri testaa arviolta vain noin 40 % piirilevyjen komponenteista. Levyllä saattaa siis olla viallisia komponentteja, vaikka levy näissä testeissä näyttäisikin toimivan odotetulla tavalla.

Passiivisia komponentteja, kuten vastuksia, keloja tai kondensaattoreita, on kaikista levyn komponenteista noin 70 %. Nykytilanteessa näitä ei siis testata lainkaan. Mikäli ladontakone osaisi testata nämä jo ladontavaiheessa, tulisi silloin jo testattua 70 % koko levyn komponenteista. Sama tapahtuisi tietenkin ladontakoneen jälkeen sijoitetulla Flying Probe -testerillä. Tällöin tuotantolinjalla on yksi vaihe lisää, mikä lisää kokonaistestiaikaa ja siten kustannuksia. Samasta syystä Flying Probella ei myöskään olisi tarkoituksenmukaista testata jokaista komponenttia erikseen, vaan vain kriittisimmät kohteet. Jos Flying Probella haluaisi löytää kaikki virheet ja testata jokaisen komponentin, testauksesta ja testaustajasta tulevat kustannukset nousisivat huomattavasti.

Nykymuotoinen testeri tässä työssä esiteltyjen muutosten jälkeen täydennettynä esimerkiksi Flying Probe -testilaitteistolla parantaisi selvästi testikattavuutta. Toiminnallisissa testeissä tulee testattua arviolta 40 % koko levyn komponenteista. Jos oletetaan, että kaikki tai ainakin enin osa passiivisista komponenteista on jo testattu, ja lisätään siihen määrä, joka saadaan toiminnallisissa testeissä, lopputulemaksi voidaan saada jopa 80–90 %:n testikattavuus. Mikäli testijärjestelmään ei lisätä erillistä passiivisten komponenttien testaajaa, täytyy piirilevyllä kehittää uusia testitapahtumia, jotta saataisiin katettua isompi osa komponentteja. Näitä on kuitenkin käytännössä mahdoton nykyisellä laitteistolla mitenkään testata. Tässä työssä esitellyissä testeissä voidaan varmistaa vain tiettyjen piirien toimivuus, koska kaikkia passiivisia komponentteja ei voida toiminnallisilla testeillä kattaa.

Piirilevyjen yläpinnoilla olevien komponenttien testit näkyvät liitteistä 6 ja 7. Niissä on merkitty eri väreillä ne komponentit, jotka tulevat kussakin testissä testattua. Nämä on käytännössä kaikki erilaisia muisti- tai suoritinpiirejä, jotka voidaan testata tässä työssä esitellyillä toiminnallisilla testeillä. Kuitenkin valtaosa, esimerkiksi piirilevyn alapuoli lähes kokonaan, on passiivisia komponentteja, joita ei käytännössä voida nykyisellä kokoonpanolla kutakin erikseen testata.

6.6 Suunnitelman toteutus

Tässä työssä esiteltyä suunnitelmaa ei todennäköisesti toteuteta sellaisenaan, vaan siitä poimitaan parhaat palat, joita yhdistellään yrityksen muiden suunnittelijoiden kanssa heidän ideoihinsa siten, että muutosten kustannukset suhteessa niistä saatuihin hyötyihin ovat järkevät. Tarkoitus olisi, että suunnitelman isot piirteet olisi toteutettu syksyyn 2012 mennessä.

Suunnitelman toteuttamisessa aikaa vievin osuus tulee olemaan ohjelmointityö sekä jigin uudelleensuunnittelu. Käytännön työ ei vaatine kovin suuria ponnisteluja, koska kaikki tarvittava laitteisto on jo olemassa. Mahdolliset lisenssihankinnat voivat kestää viikkoja, jopa kuukausia. Tähän joudutaan, jos kehitysympäristöksi valitaan LabView tai jokin muu lisenssin alainen ohjelmisto.

Testeriä voidaan käytännössä kehittää loputtomiin. Pienillä ohjelmistoparannuksilla testiä saadaan lyhennettyä sekunti sekunnilta. Tällaiseen ei kuitenkaan kannattane ryhtyä, koska optimoinnissa säästetty aika on menetetty optimointiin kuluneessa ajassa ja kustannuksissa. Ainakin kannattaa ensin toteuttaa kaikki suuret linjaukset, kuten esimerkiksi käyttöjärjestelmän vaihto tai testijigin uusiminen.

7 POHDINTA

Insinööri työn tavoitteena oli tehdä suunnitelma testerin parantamiseksi niin nopeuden, luotettavuuden kuin kattavuuden osalta. Työssä saavutettiin sille asetetut tavoitteet kohtuullisen hyvin. Suunnitelmasta ilmenee, minkälaisia muutoksia ja missä laajuudessa tarvitaan. Samalla on esitelty mahdolliset ratkaisut niihin. Yksityiskohtiin ei tarkoituksella menty, vaan pyrittiin tekemään kokonaiskuva tilanteesta, jolloin saatiin helpoimmin pahimmat niin sanonut pullonkaulat paljastettua.

Opinnäytetyön teoriaosassa esitellään yleisimmät testimenetelmät ja niiden vaatimukset. Nykyistä testijärjestelmää voisi täydentää käytännössä millä tahansa teoriaosassa esitellyllä menetelmällä.

Itse työn teko sujui jouhevasti. Yritys järjesti tilat ja kulkuluvat, lisäksi etätö oli mahdollista. Työn luonne ei vaatinut jatkuvaa läsnäoloa, jolloin sitä pystyi tekemään luontevasti ja joustavasti koulunkäynnin ohella.

Kokemusta testaamisesta ja testijärjestelmistä ennen tätä työtä minulla oli koulun lisäksi yhteensä kahdeksan kuukautta RF-testaamisesta. Testaaminen oli hiukan erilaista, mutta perusrakenne ja -ajatus oli sama sekä testerin parissa työskentelyssä että sen kehittämisideoinnissa. Työtä tehdessä koulun aikana saatu teoriapohja konkretisoitui käytännössä.

Työ vastasi koulutusalaani (tietotekniikka) ja suuntautumistani (elektroniikan suunnittelu ja testaus) täydellisesti. Työn teoria samoin kuin toteutus tukivat koulussa opittua teoriaa, samalla kun tätä työtä varten sitä joutui itse omaloitteisesti syventämään.

LÄHTEET

1. Suomen standardoimisliitto SFS ry. 2011. Johdanto laadunhallintaan ja ISO 9000 -standardeihin. Saatavissa: <http://www.sfsedu.fi/www/fi/liitetiedostot/SFS/KalvosarjaoppilaitoksilleISO9000versioSFSedusivustolle.pdf>. Hakupäivä 30.3.2012.
2. Lillrank, Paul 1998. Laatuajattelu: Laadun filosofia, tekniikka ja johtaminen tietoyhteiskunnassa. Keuruu: Otava.
3. Kokkomäki, Tapani 2011. T723303 Testausmenetelmät, 3 op. Opintojakson oppimateriaali keväällä 2011. Oulu: Oulun seudun ammattikorkeakoulu, tekniikan yksikkö.
4. Crouch, Alfred 1999. Design for test for digital IC's and embedded core systems. New Jersey: Prentice-Hall Inc.
5. Burns, Mark - Roberts, Gordon 2001. An introduction to Mixed-Signal IC Test and Measurement. Oxford: Oxford University Press.
6. Wang, Francis 1991. Digital Circuit Testing: A Guide to DFT and Other Techniques. Seattle: Academic Press, Inc.
7. Prestoy, Mike 2010. EMPF. Winning the Board Testing Task: Flying Probe versus Bed of Nails. Saatavissa: <http://www.empf.org/empfasis/2010/Feb10/winning.html>. Hakupäivä 29.3.2012.
8. Flying Probe in-circuit testing. Saatavissa: http://www.radio-electronics.com/info/t_and_m/ate/flying-Probe-tester-testing.php. Hakupäivä 2.3.2012.
9. EB ToughVoIP™. Elektrobit Oy. Saatavissa: http://www.elektrobit.com/what_we_deliver/wireless/offering/tough_voip. Hakupäivä 13.2.2012.

10. NI LabVIEW Full Development System for Linux®. Saatavissa:
<http://sine.ni.com/nips/cds/view/p/lang/en/nid/2541>. Hakupäivä 21.3.2012.
11. OpenTTCN. OpenTTCN Ltd. Saatavissa: <http://www.openttcn.com/>. Hakupäivä 5.3.2012.
12. The MY100e series: Where high mix meets high speed. Saatavissa:
<http://www.perel.fi/pdf/FA/MY100e%20Brochure%20April%202011%20P-001-0230%20rev%200000.pdf>. Hakupäivä 26.3.2012.

LIITTEET

Liite 1 Lähtötietomuistio

Liite 2 Harmaala, Jaakko 2010. EB Tough VoIP Test environments HW versio 0.0.1. Elektrobit Wireless Communications Oy. (EB confidential, vain yrityksen sisäiseen käyttöön)

Liite 3 Väärälä, Kimmo 2011. EB Tough VoIP MSU Production Test Specification versio 1.9.1. Elektrobit Wireless Communications Oy. (EB confidential, vain yrityksen sisäiseen käyttöön)

Liite 4 Väärälä, Kimmo 2011. EB Tough VoIP MCU Production Test Specification versio 1.9.1. Elektrobit Wireless Communications Oy. (EB confidential, vain yrityksen sisäiseen käyttöön)

Liite 5 Elektrobit Wireless Communications Oy. MSU test summary. (EB confidential, vain yrityksen sisäiseen käyttöön)

Liite 6 Elektrobit Wireless Communications Oy. MSU PWB TOP.

Liite 7 Elektrobit Wireless Communications Oy. MCU PWB TOP.

